Modeling Risk Anticipation and Defensive Driving on Residential Roads with Inverse Reinforcement Learning

Masamichi Shimosaka[1], Takuhiro Kaneko[1], Kentaro Nishi[1]

***Abstract*— There has been extensive research on active safety systems in the ITS community in recent years that has significantly contributed to reducing traffic accidents. However, further reduction is needed, especially on residential roads, where the reduction rate of traffic accidents is still quite small. On residential roads, traffic accidents are caused primarily by pedestrians suddenly running in front of cars and by the inattention of drivers to such risks. Automatic emergency braking systems activated by pedestrian detection are not always reliable on residential roads due to physical limitations such as too short a braking distance. To overcome the limitations of current active safety management systems, we focus on risk anticipation and defensive driving, key ideas to ensure safety on residential roads. Since defensive driving requires careful deceleration in advance of barrier lines and the corners of streets, long-term driver behavior prediction is needed. In this work, we provide a new framework of modeling risk anticipation and defensive driving with inverse reinforcement learning (IRL). In contrast to conventional driver behavior models such as hidden Markov models and maximum-entropy Markov models, our framework using IRL ensures accurate long-term prediction of driver maneuvers since the IRL is based on the Markov decision process (MDP), a goal-oriented path planning framework. Because the predicted defensive driver behaviors obtained by an MDP are appropriate only when the reward functions are carefully designed, we use *inverse* reinforcement learning, where the normative behavior of expert drivers is leveraged to optimize the reward functions. In addition to the proposed formulation of defensive driving with IRL, we provide new feature descriptors for computing reward functions to represent risk factors on residential roads such as corners, barrier lines, and speed limitations. Experimental results using actual driver maneuver data over 20 km of residential roads indicate that our approach is successful in terms of providing precise learning models of risk anticipation and defensive driving. We also found that the behavior models obtained by expert/inexperienced drivers are helpful for determining the factors in risk anticipation and defensive driving.**

I. Introduction

There has been extensive research on active safety systems in the ITS community in recent years that has significantly contributed to reducing traffic accidents. However, further reduction is needed, especially on residential roads, where the reduction rate of traffic accidents is still quite small [1]. In this work, we define a residential road as a road that has a relatively small width and is generally used only by people who live near the road for traveling within the area or reaching a main road. On residential roads, car accidents are caused primarily by pedestrians suddenly running in front of cars and by the inattention of drivers to such risks. Automatic emergency braking systems activated by pedestrian detection are not always reliable on residential roads due to physical limitations such as too short a braking distance. To overcome the limitations of current active safety management systems, we focus on risk anticipation and defensive driving, key ideas to ensure safety on residential roads. In other words, we promote the idea that drivers on residential roads should anticipate potential risks. Although there has been some research that considers potential risks on roads, including studies on pedestrian perception by vehicle-to-pedestrian communications [2], [3], there has been little focus on the idea of defensive driving based on the anticipation of risk. Modeling risk anticipation and defensive driving is helpful in terms of developing active safety systems such as an alert system based on a defensive driving model. Thus, we model defensive driving by skilled drivers and apply the model for predicting defensive driving. It is obvious that modeling defensive driving on residential roads is difficult due to the many uncertainties stemming from pedestrians. For example, it is not enough to merely follow traffic rules such as speed limits and traffic signs on residential roads, and there are no clear norms on when and how to change driving behavior. For dealing with environmental uncertainties, diversities, and ambiguous norms, a machine learning-based approach would be more appropriate than history-based approaches or approaches assuming an explicit model.

There has been some research on driving behavior modeling using machine learning techniques. For example, behavior prediction using a dynamic Bayesian network was able to forecast a driver maneuver a few seconds later [4]. However, prediction in a few seconds is not sufficient for active safety systems on residential roads, and predicting behaviors over a longer time range is needed in terms of defensive driving. A machine learning-based method for predicting turns at intersections has also been presented [5]. However, this method was not sufficient for predicting next turns when it came to active safety on residential roads. In other words, it is preferable to model driving behavior in a scene that requires a balance of comfortable speed and defensive speed, which is obviously much slower than the legal speed limit, which is 30 km/h in Japan. That is, it is especially important to model deceleration maneuvers based on the anticipation of risk.

In this work, we represent driving behaviors as a sequence of decisions of acceleration/deceleration and states of position and velocity with a Markov decision process (MDP). In an MDP, given the reward in each state, relatively

[1]M. Shimosaka and K. Nishi are, and T. Kaneko was with the Department of Mechano-Informatics, Graduate School of Information Science and Technology, the University of Tokyo, Tokyo 113-8656, Japan. {simosaka, kaneko, nishi}@ics.t.u-tokyo.ac.jp

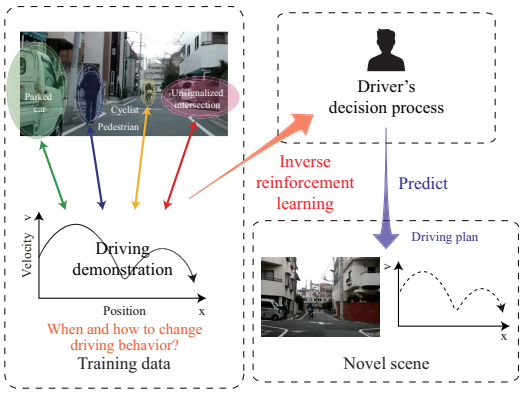


Fig. 1. The concept of our work. We learn risk anticipation and defensive driving on residential roads with inverse reinforcement learning. In a novel scene, an optimal driving plan is predicted using the learned model.

long-term driving behavior can be predicted by planning an optimal state sequence toward the goal, incorporating both an immediate reward and expected future rewards. However, it is not trivial to design appropriate reward functions for representing defensive driving. Therefore, we have to handle the inverse problem, that is, an approach to learning the reward function that represents the model of decision making from actual driving demonstrations. As a first step towards a practical risk-sensitive driving model, we focus on acceleration/deceleration on residential roads. Our modeling framework with inverse reinforcement learning (IRL) is shown in Fig. 1. By using the trained model with actual driving demonstrations, long-term driver behavior can be predicted even in novel scenes.

In our experiment, we acquired actual driving data on residential roads in Japan. The results indicate that our approach is successful in terms of providing precise learning models of risk anticipation and defensive driving. We also applied our approach to data from an inexperienced driver and found that our approach could successfully extract the environmental factors to be focused on in defensive driving by comparing the skilled driver model with the inexperienced driver model.

The contributions of this paper are as follows. 1) We organized the requirements for machine learning-based driving behavior modeling on a residential road, which has rarely been the target in previous research. 2) We built a novel driver behavior modeling framework with inverse reinforcement learning to provide accurate long-range driver behavior. 3) We designed feature descriptors based on geographical information. 4) We acquired data of driving behaviors on actual residential roads and extracted environmental factors to be focused on in defensive driving by comparing an expert driver model with an inexperienced driver model.

The rest of this paper is organized as follows. In section II, we discuss related work. Section III presents our model formulation of the risk anticipation and defensive driving and optimization method. In section IV, we describe environmental features, and in section V, we describe the experiments we performed to verify our model. We conclude with a brief summary in section VI.

II. Related Work

Related work in terms of situations for active safety for automobiles and driving behavior modeling is briefly discussed in this section. Specifically, we describe target situations of existing research related to active safety systems and discuss existing approaches to driving behavior modeling.

A. Target Situations

Target situations are classified roughly into highways, urban streets, and residential roads in terms of the level of difficulty of driving behavior modeling. Although residential roads can also be urban streets, here we define a residential road as a road in which there is a risk of crashing into pedestrians stemming from the existence of both pedestrians and vehicles.

Highways are tightly structured, are accessible only by vehicles, and have no intersections. Therefore, driving behavior modeling is relatively easy on highways. Studies on lane changes [6], [7] and driving at exits [8] are examples of research related to highway scenarios.

Urban streets differ from highways in that they have intersections. Many car accidents occur at intersections on urban streets, and so there has been a lot of research on how to avoid crashes at intersections, e.g., [5], [9]. Car-following behavior models have also been researched [10], [11] to address traffic jams at unsignalized intersections on urban streets.

On residential roads, in contrast to urban streets, we have to consider the potential risks of pedestrians and cyclists as well as vehicle interactions. Pedestrian perception via vehicle-to-pedestrian communication [2], [3] is one solution to avoid potential risks on residential roads. Nevertheless, it is still essential to perform defensive driving and to anticipate potential risks. From this aspect, it should be noted that there has been very little research that focuses on defensive driving itself.

B. Modeling Methods

Approaches to driving behavior modeling can be classified into three types: matching the current scene with previously observed data [12], using simulation based on explicit models [9], and taking machine learning approaches to deal with uncertain scenes and driving behaviors [4], [5].

Approaches using matching with previously observed data [12] enable long-term prediction in known scenes where data have been previously obtained. However, it can be difficult to apply this approach to novel scenes. The approach using an explicit model [9] can be used without data obtained previously, but we still have to consider all possible scenarios that may happen in a real situation when making this model.

Approaches using machine learning are attractive because they can deal with uncertain environments and driving behaviors by using stochastic analysis based on previous data. As mentioned in section I, we have to tackle the ambiguities inherent in defensive driving on residential roads.

Therefore, an approach using machine learning is appropriate for modeling on residential roads. There has been some research on driving behavior modeling with machine learning. For example, there is a method that predicts car-following behaviors and lane changes on a highway based on present scenes such as the position of other vehicles by using a dynamic Bayesian network [4]. A hidden Markov model-based method proposed by H. Berndt and K. Dietmayer predicts turns at intersections on urban streets [5]. Most common machine learning-based approaches based on Markov-based assumption and location history can predict short-term behavior, such as behaviors occurring in the next few seconds or the next discrete step. Moreover, their target scenarios are often tightly structured environments such as highways, and the prediction targets are relatively rough behaviors such as turns and lane changes. It should be noted that the modeling of long-term acceleration/deceleration behavior on residential roads has rarely been the target in previous research.

III. Model Formulation and Optimization

As stated in section I, our modeling target is decision making in acceleration and deceleration. Our modeling assumes that a global route is known (that is, we are not concerned with route planning), since route searching and destination estimation are becoming an active area of research [13], [14] and navigation systems currently enjoy widespread use. Also, in residential areas, residents usually drive the same route every day. Under this assumption, we target driving behavior in linear segments, as shown in Fig. 2. The segment starts with a turn or stop line and ends with the next turn or stop line. The driving maneuver of acceleration and deceleration in a segment is considered a unit of behavior.

In this work, we model defensive driving with a Markov decision process (MDP) that incorporates the dynamics of decision-making into a Markov process. Fig. 3 shows underlying graphical model for an MDP. In an MDP, given reward function $R(s)$, we can plan the optimal state sequence using dynamic programming [15] to incorporate both an immediate reward and an expected future reward. Therefore, an MDP enables long-term prediction and is appropriate for defensive driving modeling. Note that the predicted defensive driver

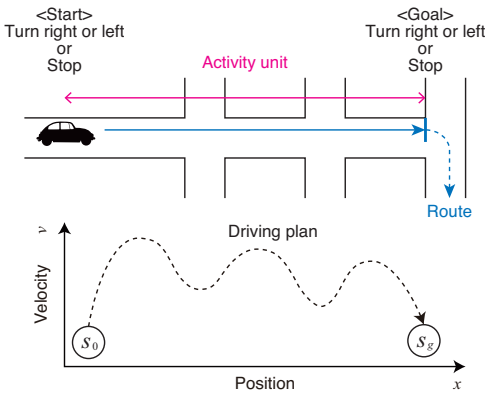


Fig. 2. Assumed situation and target driving behavior.

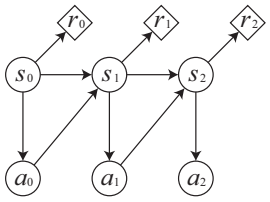


Fig. 3. Underlying graphical model for an MDP.

behaviors obtained by an MDP are appropriate only when the reward functions are designed carefully.

It is quite easy to obtain the optimal path in an MDP when the reward function is fixed, but it is extremely difficult to design the reward function appropriately in the first place. Therefore, in this work, we address the inverse problem—namely, we learn the optimal reward function from an actual driving demonstration. This inverse problem is able to be solved by inverse reinforcement learning (IRL) [16], [17], [18].

A. Model Representation with Markov Decision Process

In the behavior unit defined above, a driver should perform acceleration and deceleration while looking ahead to a goal, i.e., the next turn or stop. Therefore, we formulate defensive driving as a planning problem from the current state to a goal state in position-velocity space with an MDP. This formulation represents the driver's action selection sequence to the goal. We represent state s as a combination of position x and velocity v as $s = (x, v)$. Both state s and action a are discretized in a certain way (described in detail in section V). We represent the dynamics of driving behaviors with discrete states and actions, defining state transition probability $P(s'|s, a)$.

To make the connection between environmental factors $\boldsymbol{f}(s)$ and reward function $R(s|\boldsymbol{\theta})$, the reward function is assumed to be represented as $R(s|\boldsymbol{\theta}) = \boldsymbol{\theta}^{\mathsf{T}}\boldsymbol{f}(s)$, where each $f_k(s)$ is a feature based on an environmental factor that may affect the driving behavior and $\boldsymbol{\theta} \geq \mathbf{0}$ denotes a parameter of weights. This means that $R(s|\boldsymbol{\theta})$ is represented as a weighted combination of features $\boldsymbol{f}(s) = [f_1(s)...f_K(s)]^{\mathsf{T}} \leq \mathbf{0}$. The details of $f_k(s)$ are described in section IV. The likelihood for state sequence $\zeta = \{(s_0, a_0), (s_1, a_1), ...\}$ is represented as [14]:

$$P(\zeta|\boldsymbol{\theta}) = \frac{1}{Z(\boldsymbol{\theta})} \exp\left(\sum_t (\boldsymbol{\theta}^{\mathsf{T}}\boldsymbol{f}(s_t) + \log P(s_{t+1}|s_t, a_t))\right) \tag{1}$$

where $Z(\boldsymbol{\theta})$ is a normalizing function.

B. Planning in an MDP with Dynamic Programming

In an MDP, given initial states $P(s_0)$, transition probability $P(s'|s, a)$, and reward function $R(s)$, we can predict a state and action sequence $\zeta = \{(s_0, a_0), (s_1, a_1), ...\}$ with dynamic programming [14]. As described next, we can obtain optimal policy $\pi(a|s)$ by using a backward pass and can predict state sequence ζ and obtain the expected state

visitation count $D(s_i)$ of s_i by performing state transition $s \to s'$ according to policy $\pi(a|s)$ with a forward pass.

1) Backward pass: Let weight parameter $\boldsymbol{\theta}$ be determined. At this time, we compute state log partition function $V^{\mathrm{soft}}(s)$ and state-action log partition function $Q^{\mathrm{soft}}(s,a)$ with a backward pass so that the reward function of the final state becomes $\phi(s)$.

As stated earlier, in this study, we assume the behavior unit to start with a turn or stop line and to end with the next turn or stop line. The final state means the state with low velocity at an intersection or stop line. The final state is obviously not always the same because actual humans are performing the driving behavior. Therefore, we consider Gaussian kernel $P_g(s)$ with the center of goal state $s_g = (x_g, v_g)$ for state $s = (x, v)$ and represent the reward function at the final state as $\phi(s) = \log(P_g(s))$, where the goal state s_g is represented as $s_g = (x_g, v_g)$ with goal position x_g and minimum velocity v_g.

Intuitively, the backward pass evaluates the expected reward from all states to the goal state. State log partition function $V^{\mathrm{soft}}(s)$ is a soft estimation of the expected reward obtained when reaching the state near s_g from state s, and the state-action log partition function $Q(s,a)$ is a soft estimation of the expected reward obtained when reaching the state near s_g after performing action a at state s. After $V^{\mathrm{soft}}(s)$ and $Q^{\mathrm{soft}}(s,a)$ converge, the policy computed by $\pi_\theta(a|s) = \exp(Q^{\mathrm{soft}}(s,a) - V^{\mathrm{soft}}(s))$.

2) Forward pass: Forward pass is used to compute $D(s)$, which is the expected state visitation count of state s. $D(s)$ is computed by propagating initial state $P_0(s)$ according to policy $\pi_\theta(a|s)$ computed with the backward pass described above. Probability propagation is prevented by setting $D(s) = 0$ after goal state s_g in implementation, otherwise, the propagation continues after goal state s_g.

C. Training with Inverse Reinforcement Learning

In the learning step, we optimize weight parameter $\boldsymbol{\theta}$ by minimizing negative log likelihood $-L(\boldsymbol{\theta})$ with regularization term $\Omega(\boldsymbol{\theta})$ and then compute optimal policy $\pi(a|s)$. As the regularization term, we use L1 regularization $\Omega(\boldsymbol{\theta}) = \lambda \sum_i |\theta_i|$ for feature selection. We first compute optimal weight parameter $\boldsymbol{\theta}^*$ by minimizing objective function $-L(\boldsymbol{\theta}) + \Omega(\boldsymbol{\theta})$, as

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \{-L(\boldsymbol{\theta}) + \Omega(\boldsymbol{\theta})\} \quad (2)$$

$$= \underset{\boldsymbol{\theta}}{\operatorname{argmin}} \left\{ -\sum_{\tilde{\zeta}_i \in \mathcal{D}} \log P(\tilde{\zeta}_i|\boldsymbol{\theta}) + \lambda \sum_i |\theta_i| \right\}, \quad (3)$$

where $\mathcal{D}$ denotes a dataset of M sequences of a driver's demonstration data are represented as $\mathcal{D} = \{\tilde{\zeta}_1, ..., \tilde{\zeta}_M\}$ with $\tilde{\zeta}_i = \{(\tilde{s}_{i,0}, \tilde{a}_{i,0}), ..., (\tilde{s}_{i,T_i}, \tilde{a}_{i,T_i})\}$. The gradient of log likelihood $\nabla L(\boldsymbol{\theta})$ is formulated as

$$\nabla L(\boldsymbol{\theta}) = \tilde{\boldsymbol{f}} - \sum_{\zeta} P(\zeta|\boldsymbol{\theta}) \boldsymbol{f}(\zeta) = \tilde{\boldsymbol{f}} - \sum_{s_i} D(s_i) \boldsymbol{f}(s_i), \quad (4)$$

where $\tilde{\boldsymbol{f}}$ is an expected empirical feature count represented as $\tilde{\boldsymbol{f}} = \frac{1}{M} \sum_i \boldsymbol{f}(\tilde{\zeta}_i)$. The learned model is also used to extract the environmental factors that affect defensive driving by obtaining non-negative weight parameters corresponding to the features of the environmental factors. For this reason, we use exponentiated gradient descent: that is, we compute optimal weight parameters by repeating $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} \exp(\eta(\nabla L(\boldsymbol{\theta}) - \nabla \Omega(\boldsymbol{\theta})))$ with step width η. $D(s_i)$ is computed with backward pass and forward pass, as described above. Once the weight parameter is determined, we can use the two algorithms to predict driving behaviors, as well.

IV. Designing Feature Descriptors

We based the feature descriptors on road configuration and traffic signs, as these two items have an effect on the potential risks inherent in driving on residential roads. These environmental factors are assumed to be prospectively known by predriving because they never change. We focus our attention on intersections, the blind corners near intersections, and the positions of the start and goal, as shown in Fig. 4 (a), where red and blue lines respectively indicate the position-velocity space of an expert driver and an inexperienced driver in an actual driving demonstration. The positions of intersections, blind corners, the start, and the goal are annotated with vertical lines.

We extract five kinds of features from these geographical factors. An example of each is shown in Figs. 4 (b)(f), where the blue region indicates a low reward and the red region indicates a high reward. Intuitively, the red region is more likely to be passed in driving behaviors. The value at s is used as feature $f(s)$ and the final reward function $R(s)$ is learned as a weighted combination of these features with sparse weight parameters, as described in section III. All features are represented with Gaussian kernels. For example, the feature related to velocity repression at start position x_s is generated as $f(s) = -\exp\left(-(\boldsymbol{s} - \boldsymbol{s}_s)^{\mathsf{T}} \Sigma^{-1} (\boldsymbol{s} - \boldsymbol{s}_s)\right)$, where $\boldsymbol{s} = [x, v]^{\mathsf{T}}$ is a vector corresponding to $s = (x, v)$ and $\boldsymbol{s}_s = [x_s, v_{\max}]^{\mathsf{T}}$ is a vector corresponding to start position x_s and the speed limit $v_{\max}$. Σ is the covariance matrix. We generate multiple features with different widths of kernels by changing Σ and then determine the optimal width by obtaining sparse weights $\boldsymbol{\theta}$ using learning with the L1 regularization term described above. Note that these features can be computed online using GPS and map data since the only information required is the position of each environmental factor.

a) Velocity repression at start and goal: We generate features related to start position, goal position, and maximum speed. This is based on the observation that skilled drivers slow down the velocity near the start and goal. Fig. 4 (b) shows an example with a certain width.

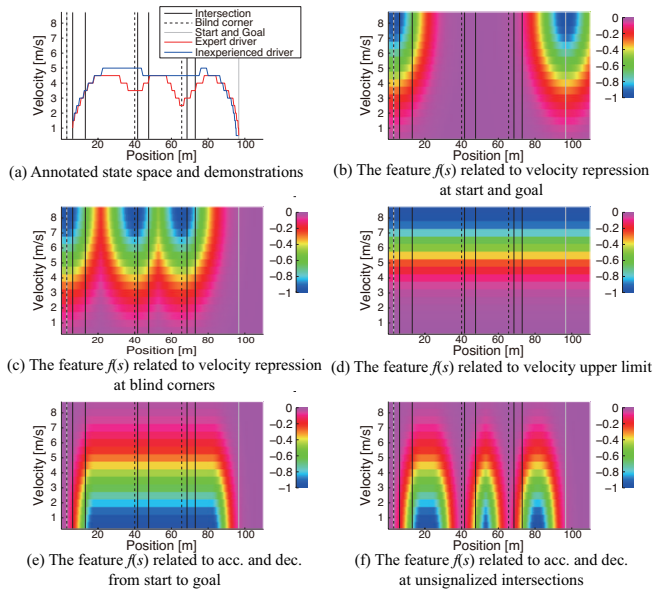
b) Velocity repression at blind corners near unsignalized intersections: Defensive drivers may slow down the speed before a blind corner near unsignalized intersections. Fig. 4 (a) shows the expert driver finishing deceleration at the position of the blind corners. We generate features related to this point, an example of which is shown in Fig. 4 (c).

c) Features related to velocity upper limit: We assume drivers obey the legal speed limit over the entire road. To

represent this, we generate features according to the distance from the upper limit of velocity. An example of this feature is shown in Fig. 4 (d).

d) Features related to acceleration and deceleration from start to goal: To represent acceleration and deceleration from start to goal, we generate a feature whose distribution varies according to the distance from the start and the goal position in the region near from the start and goal. The distribution is constant in the region far from the start and goal, as shown in Fig. 4 (e).

e) Features related to acceleration and deceleration at intersections: To represent acceleration and deceleration related to intersections, we generate features whose distribution changes according to the distance from intersections. An example is shown in Fig. 4 (f).



(a) Annotated state space and demonstrations

(b) The feature $f(s)$ related to velocity repression at start and goal

(c) The feature $f(s)$ related to velocity repression at blind corners

(d) The feature $f(s)$ related to velocity upper limit

(e) The feature $f(s)$ related to acc. and dec. from start to goal

(f) The feature $f(s)$ related to acc. and dec. at unsignalized intersections

Fig. 4. Annotated state space, demonstrations, and extracted features.

V. Experiments

A. Experimental Setup

The experimental vehicle we set up for acquiring actual driving data is shown in Fig. 5. A LIDAR, a GPS sensor, and cameras are attached to the experimental vehicle. The data are used to extract the features described in section IV. The position is calculated by accumulating the velocity data obtained via a controller area network (CAN) bus.

We selected four courses on residential roads in Tokyo, Japan. Each course starts with a turn or stop line and ends with the next turn or stop line. Each course contains two or three unsignalized intersections. The distance and width of the whole course, the distances between intersections, and the sizes of the intersections are unique to each course.

We selected two drivers as subjects: one who is an expert driver working as a taxi driver and the other who is an inexperienced driver who drives only a few times per year. The total travel distance of the two drivers was about 20 km and each passed through roughly 200 unsignalized intersections. We performed the experiments based on a leave-one-out validation, namely, we used three courses as training data and the rest as test data, and repeated all four combinations.

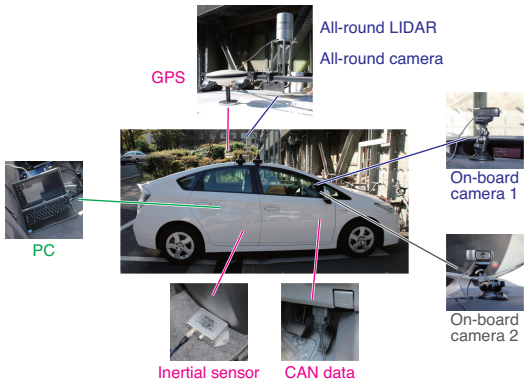


Fig. 5. Experimental vehicle and setting sensors.

B. Implementation

The state space of the MDP in this work is discrete space, as shown in Fig. 6. Let the current space $s = (x, v)$. The next step is then $s_a = (x + v + 1, v + 1)$ if the driver accelerates, $s_m = (x + v, v)$ if the driver maintains the speed, and $s_d = (x+v-1, v-1)$ if the driver decelerates. Thus, $P(s_{t+1}|s_t, a_t)$ is defined in a deterministic manner. We discretize the state at 0.5 m/s intervals into 17 steps in velocity from 0.5 m/s $=$ 1.8 km/h to 8.5 m/s $= 30.6$ km/h. This covers the range from walking speed (4.0 km/h) to legal maximum speed (30.0 km/h). We discretize the time at 5 Hz intervals. Generally speaking, drivers need about one second to start braking after detecting risk. The discretization enables prediction in a shorter time than one second. Position is discretized from 0.5 m/s $\times$ 0.2 s $= 0.1$ m to 8.5 m/s $\times$ 0.2 s $= 1.7$ m with these discretizations. Namely, 0.1 m is one distance unit and the position changes by 1-17 units according to the velocity in one step.

In the experiment, the positions of the intersections were manually annotated with 3D point cloud data from LIDAR. Much research on detecting various objects has recently been performed [19], and we can combine these techniques for the automatic extraction of environmental features.

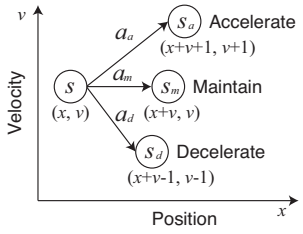


Fig. 6. State and action representation.

C. Evaluation Metric

We use modified Hausdorff distance (MHD) [20] as the metric to evaluate similarity between the state sequence of the actual driving demonstration and the state sequence generated with learned policy $\pi(a|s)$ in the position-velocity space. MHD is an extension of Hausdorff distance that enables the matching of time-series data. MHD represents the distance between time-series data $P = \{p_t\}_{0 \le t < T_p}$ and $Q = \{q_t\}_{0 \le t < T_q}$ as

$$h_\alpha(P, Q) = \mathrm{ord}^\alpha_{p \in P} \left(\min_{q \in N(C(p))} d(p, q) \right), \tag{5}$$

where $N(q)$ denotes the set of neighbor points to point q in Q and $C(p)$ denotes a point q in Q related to p in data sequence P. $\mathrm{ord}^\alpha_{p \in P} f(p)$ is the value of $f(p)$ below which the α of the values may be found. Since this is a directed metric, we use $H_\alpha(P, Q) = \max(h_\alpha(P, Q), h_\alpha(Q, P))$ for the evaluation as an undirected metric. We compute the MHDs between state sequence P in actual demonstration and 100 state sequences obtained by random sampling with the learned policy $\pi(a|s)$ from starting state. We use the average of the MHDs for evaluation. We set α, a parameter of MHD, as $\alpha = 0.5, 0.9$. Note that when $\alpha = 0.5$, the MHD represents the median distance of the sequences, and when $\alpha = 0.9$, the MHD represents the 90 percentile in order of increasing. From now, we write them as MHD_{50} and MHD_{90}, respectively.

D. Compared Methods

We use the location-based Markov model (LBMM) and the maximum-entropy Markov model (MEMM) as comparative methods.

1) Location-Based Markov Model: The location-based Markov model (LBMM) is a history-based method that does not use any features. It computes policy $\pi(a|s)$ from observed action in the training set according to locations. With this model, first, we divide the roads into four regions: l_s, which is the nearest region to the start position, l_b, which is the nearest region to an intersection position and start side from the intersection, l_a, which is the nearest region to an intersection and the goal side of the intersection, and l_g, which is the nearest region to the goal. We calculate l_c, which is the region of current state s, and determine policy $\pi(a|s)$ as $\pi(a|s) \propto c_{l_c}(a, s_{l_c}) + \alpha$, where state s_{l_c} is represented by $s_{l_c} = (x_{l_c}, v)$, x_{l_c} denotes the distance from the reference point of l_c, $c_{l_c}(a, s_{l_c})$ is the count at which the action a is observed in state s_{l_c}, and α is a pseudo count determined using cross-validation.

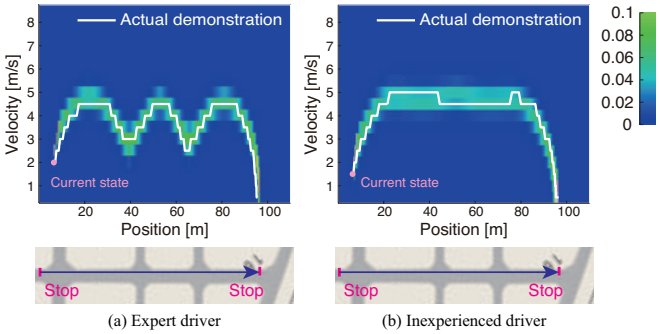
2) Maximum-Entropy Markov Model: With the maximum-entropy Markov model (MEMM), the policy is computed by $\pi(a|s) \propto \exp\{\boldsymbol{w}_a^\mathsf{T} \boldsymbol{F}(s)\}$, where $\boldsymbol{F}(s)$ is a vector of features for the neighbor states of current state s.

We use the features for all six possible states at the next step and the previous step in addition to the features for the current state s. That is, we incorporate the features for all of the seven states in this model.

Although our proposed method selects the optimal action looking ahead to the goal incorporating immediate reward and expected future rewards, MEMM incorporates the features only for the next and previous steps. Note that it is intractable to incorporate all features towards the goal in MEMM because we would have to compute the features for all possible state sequences from the current state to the goal state, which is not feasible.

E. Experimental Results

We conducted experiments to determine how well our method could model defensive driving. Driving behaviors were modeled using data from an expert driver and an inexperienced driver. None of the data used contained any dynamic environmental changes. The modeling results are shown in Fig. 7, where the background color indicates $D(s)$, which is the expected state visitation count from current state using learned policy $\pi(a|s)$. A lighter background color indicates a higher $D(s)$. The white lines show the actual demonstrated maneuvers of the expert driver and the inexperienced driver and the map below corresponds to the position of the upper figure. The white lines are well accorded with the lighter regions, and the expert driver diminished the velocity before passing the intersections (Fig. 7(a)); the inexperienced driver, however, did not. These results imply that our approach is successful in terms of providing precise learning models of risk anticipation and defensive driving. Also, the difference between the two drivers is helpful in terms of developing an active safety system such as an alert system for inexperienced drivers.

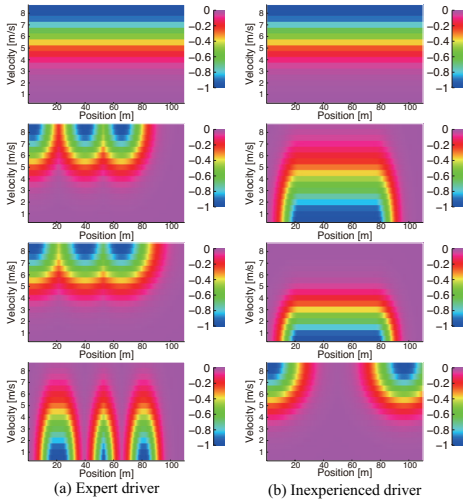


(a) Expert driver (b) Inexperienced driver

Fig. 7. Predictions of future visitation expectations given current states and policies. Maps cited are from Google Maps [21].

Fig. 8 shows highly weighted features when data on the Course 1 is used as the test data. The top four features are shown. Fig. 8 (a) shows the model of the expert driver where, beginning at the top, the feature related to velocity upper limit, the two features related to velocity repression at blind corners near unsignalized intersections, and the feature related to acceleration and deceleration at unsignalized intersections are shown, and Fig. 8 (b) shows the model of the inexperienced driver where, beginning at the top, the feature related to velocity upper limit, the two features related to velocity repression at start and goal, and the feature related to acceleration and deceleration from start to goal are shown. The features related to unsignalized intersections are highly weighted in the expert driver model compared with

that of the inexperienced driver, indicating that the expert driver was more likely to perform defensive driving while considering potential risks at unsignalized intersections. This demonstrates that our model is also useful for extracting which environmental factors to focus on with defensive driving by examining highly weighted features.



(a) Expert driver (b) Inexperienced driver

Fig. 8. Highly weighted features.

Table I lists the qualitative results of the expert driver's model compared to other approaches using MHD, which represents similarity between the state sequence of actual driving demonstration and the state sequence generated with learned policy $\pi(a|s)$. The values indicate mean MHDs and their standard deviations of all the courses. The results show that the feature-based methods (the proposed method and MEMM) outperform LBMM. Though the proposed method and MEMM show comparative performances, MEMM may have the label bias problem when test data have exceptional events stemming from uncertainty factors such as pedestrians suddenly running in front of cars. Our proposed method would deal with this problem since it is goal-oriented method. Further experiments are needed in order to confirm the performence against uncertainty factors.

TABLE I

COMPARISON WITH DIFFERENT METHODS.

Method	MHD_{50}	MHD_{90}
LBMM	3.189 ± 0.572	6.617 ± 0.749
MEMM	0.836 ± 0.039	1.540 ± 0.099
Proposed	$\mathbf{0.879 \pm 0.042}$	$\mathbf{1.477 \pm 0.111}$

VI. CONCLUSION

We proposed an approach for modeling risk anticipation and defensive driving based on actual driving demonstration data and environmental factors using inverse reinforcement learning for active safety systems on residential roads. Experimental results using actual driver maneuver data on residential roads demonstrate that our approach is successful in terms of providing precise learning models of risk anticipation and defensive driving. Our method achieves comparative performance among state-of-the-art methods. The results also show that our approach enables us to extract environmental factors on which to focus in defensive driving from model parameters by comparing a skilled driver's model with an inexperienced driver's model. Our future work will include large-scale experiments with a wide range of drivers, areas, and times. To make our approach more practical, online implementation of driver behavior prediction with inexpensive and reliable sensors as well as extension to practical scenes including pedestrians and bicycles where redesigned feature descriptors for such moving objects as pedestrians and bicycles would be useful.

ACKNOWLEDGEMENTS

We would like to sincerely thank Mr. Tokuya Inagaki of DENSO Corporation for his deployment of driver behavior logging systems and for constructive comments and helpful suggestions on making the experiments more feasible.

REFERENCES

[1] *ITARDA INFORMATION (in Japanese)*, No. 98, Institute for Traffic Accident Research and Data Analysis, 2013. [Online]. Available: http://www.itarda.or.jp/itardainfomation/info98.pdf
[2] D. Westhofen *et al.*, "Transponder- and Camera-Based Advanced Driver Assistance System," in *Proc. of IV2012*, pp. 293–298.
[3] M. Liebner *et al.*, "Active Safety for Vulnerable Road Users based on Smartphone Position Data," in *Proc. of IV2013*, pp. 256–261.
[4] T. Gindele *et al.*, "A Probabilistic Model for Estimation Driver Behaviors and Vehicle Trajectories in Traffic Environments," in *Proc. of ITSC2010*, pp. 1625–1631.
[5] H. Berndt and K. Dietmayer, "Driver intention inference with vehicle onboard sensors," in *Proc. of ICVES 2009*, pp. 102–107.
[6] W. Yao *et al.*, "Lane Change Trajectory Prediction by using Recorded Human Driving Data," in *Proc. of IV2013*, pp. 430–436.
[7] D. Marinescu *et al.*, "On-ramp Traffic Merging using Cooperative Intelligent Vehicles: A Slot-based Approach," in *Proc. of ITSC2012*, pp. 900–906.
[8] S. Hold *et al.*, "ELA - an Exit Lane Assistant for Adaptive Cruise Control and Navigation Systems," in *Proc. of ITSC2010*, pp. 629–634.
[9] M. Liebner *et al.*, "Driver Intent Inference at Urban Intersections using the Intelligent Driver Model," in *Proc. of IV2012*, pp. 1162–1167.
[10] K. Lidström and T. Larsson, "Model-based Estimation of Driver Intentions Using Particle Filtering," in *Proc. of ITSC2008*, pp. 1177–1182.
[11] P. Anglitirakul *et al.*, "Evaluation of Driver-Behavior Models in Real-World Car-Following Task," in *Proc. of ICVES2009*, pp. 113–118.
[12] C. Hermes *et al.*, "Long-term Vehicle Motion Prediction," in *Proc. of IV2009*, pp. 652–657.
[13] J. Krumm and E. Horvitz, "Predestination: Inferring destinations from partial trajectories," in *Proc. of Ubicomp2006*, pp. 243–260.
[14] B. D. Ziebart *et al.*, "Maximum Entropy Inverse Reinforcement Learning," in *Proc. of AAAI2008*, pp. 1433–1438.
[15] M. L. Puterman, *Markov Decision Process: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., 1994.
[16] A. Y. Ng and S. J. Russell, "Algorithms for Inverse Reinforcement Learning," in *Proc. of ICML2000*.
[17] P. Abbeel and A. Y. Ng, "Apprenticeship Learning via Inverse Reinforcement Learning," in *Proc. of ICML2004*.
[18] K. M. Kitani *et al.*, "Activity Forecasting," in *Proc. of ECCV2012*, pp. 201–214.
[19] T. Gandhi and M. M. Trivedi, "Pedestrian Protection Systems: Issues, Survey and Challenges," *IEEE Trans. on ITS*, vol. 8, no. 3, pp. 413–430, 2007.
[20] S. Atev *et al.*, "Learning Traffic Patterns at Intersections by Spectral Clustering of Motion Trajectories," in *Proc. of IROS2006*, pp. 4851–4856.
[21] "Google map," https://www.google.com/maps/, 2014.